UNITÉ DE RECHERCHE
INRIA-ROCQUENCOURT

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P. 105
78153 Le Chesnay Cedex
France
Tél.:(1) 39 63 55 11

Rapports de Recherche

N° 1225

Programme 5
Automatique, Productique,
Traitement du Signal et des Données

A PROPOSED CLASSIFICATION OF INTELLIGENT MANUFACTURING SYSTEMS USING CONSISTENCY ANALYSIS

Gau Rong LIANG

GROUPE DE RECHERCHE
METZ

Mai 1990

A PROPOSED CLASSIFICATION OF INTELLIGENT MANUFACTURING SYSTEMS USING CONSISTENCY ANALYSIS

PROPOSITION D'UNE CLASSIFICATION DE SYSTEMES DE PRODUCTION INTELLIGENTS UTILISANT L'ANALYSE DE CONSISTANCE

Abstract :

A new classification of Intelligent Manufacturing Systems (IMS) based on consistency analysis is proposed. Through this approach, IMS can be initially classified into Declarative Manufacturing Systems (DMS), which are absolutely consistent manufacturing systems, and Descriptive Manufacturing Systems (D'MS), which the consitency of manufacturing systems have to be checked or verified. Furthermore two main types of DMS, Functional Manufacturing Systems (F'MS) and Relational Manufacturing Systems (RMS), are analyzed in detail using lambda calculus and predicate calculus, respectively. On the other hand, the Touretzky's inheritance theory is used as an analytical tool for checking the consistency of D'MS. In summary, because different manufacturing systems use different mathematical tools, a category-theoretic foundation of intelligent manufacturing systems is conjectured for the future research.

KEY WORDS : INTELLIGENT MANUFACTURING SYSTEM, ARTIFICIAL INTELLIGENCE, CONSISTENCY ANALYSIS, LAMBDA CALCULUS, FIXPOINT SEMANTICS, TOURETZKY'S INHERITANCE THEORY, EQUATIONAL CALCULUS, CATEGORY THEORY.

Résumé :

Nous proposons une nouvelle classification des systèmes de production intelligents (Intelligent Manufacturing Systems ou IMS) basée sur l'étude de la consistance des systèmes. Nous pouvons distinguer dans un premier temps les systèmes de production déclaratifs (Declarative Manufacturing Systems ou DMS), qui sont des systèmes consistants, et les systèmes de production descriptifs (Descriptive Manufacturing Systems ou D'MS), dont la consistance doit être testée ou vérifiée.

De plus, à l'intérieur des DMS nous étudierons en détail les systèmes de production fonctionnels (F'MS) et le systèmes de production relationnels (RMS) au moyen du λ-calcul et du calcul des prédicats.

Nous utiliserons également la théorie de l'héritage de Tourensky comme un outil analytique servant à tester la consistance des D'MS. Finalement, nous conjecturons pour nos recherches futures, une approche basée sur la théorie des catégories unifiant les différents outils mathématiques utilisés par les différents systèmes de production intelligents.

MOTS CLES : SYSTEMES DE PRODUCTION INTELLIGENTS, INTELLIGENCE ARTIFICIELLE, ANALYSE DE CONSISTANCE, LAMBDA CALCUL, SEMANTIQUE A POINT FIXE, THEORIE DE L'HERITAGE DE TOURENSKY, CALCUL EQUATIONNEL, THEORIE DES CATEGORIES.

A PROPOSED CLASSIFICATION OF INTELLIGENT MANUFACTURING SYSTEMS USING CONSISTENCY ANALYSIS

Gau Rong Liang

Department of Industrial Engineering & Management,
National Chiao Tung University,
Hsinchu, Taiwan, R. O. C.

Visiting Researcher in Project SAGEP,
INRIA-Lorraine, France

INTRODUCTION

With the meaning of words, an intelligent manufacturing system is a manufacturing system which has the capability to reflect the intelligent behaviors. Currently, a major approach to achieve this goal is to model the underlying manufacturing system through artificial intelligence (AI) methods [1,2,3,4]. In general, there are two important schools in AI methods: symbolist and connectionist [5]. In this paper, the classification of IMS is analyzed with respect to the symbolist approach only.

In the following, the salient features of IMS are surveyed from the viewpoints of the Turing Test and of the mathematical tools which are of use for analyzing IMS, at first. A comparison of mathematical tools is done with other manufacturing systems such as flexible manufacturing system, computer-integrated manufacturing, etc. Second, the reason of using consistency analysis is explained. From this viewpoint, two kinds of systems are classified out: F'MS and RMS. Third, the RMS concept is applied to machine tool example using predicate calculus. Fourth, the F'MS concept is applied to AMRF example using lambda calculus. Fifth, a classification of DMS is concluded under the concepts of RMS and F'MS. Sixth, the Touretzky's inheritance theory is used as an analytical tool for D'MS. Finally, a category-theoretic foundation of IMS is conjectured.

FEATURES OF INTELLIGENT MANUFACTURING SYSTEMS

Two salient features of IMS are discussed here. One is the famous Turing Test [6,7]. Turing suggested a test in which we have typewriter communication to two rooms. There are a man in one room and a woman in the other room. Both the man and the woman would claim to be a woman through the typewriter, and it would be our problem to decide who is telling the truth. Instead, Turing suggested we could have a person in one room and a computer in the other, both claiming to be a person, and we would have to decide which is the truth. If we failed at this, then, say, the computer reflects the intelligence.

Essentially, the digital communication way is a constraint on the expressibility of human intelligence. For example, only discrete events can happen on this communication channel. Although some discrete approach can approximate non-isomorphic events in equivalence like nonstandard analysis [8], it is difficult to say this approximation can be always done for all possible events. From this viewpoint, the Turing Test practically reflects the intelligence in discrete way. However, we call this "the intelligence in Turing way". In automated factories, this is an important concept in IMS because many machines are controlled in Turing way.

The other is the mathematical tools which can be applied to analyzing IMS. Four categories of computerized manufacturing systems are compared as follows: Direct Digital Control (DDC), Flexible Manufacturing System (FMS), Computer-Integrated Manufacturing (CIM), and IMS. Basically, the DDC technology emphasizes the control of a set of points such as the PID controllers. The performance indices are stability, fast response, etc. The FMS technology emphasizes the flexibility of changing products. The major performance indices are utility and efficiency. The CIM technology emphasizes the integration of each subsystem. The major effort is to find good protocols or standards in order to coordinate each subsystem. The IMS technology emphasizes the aspect of intelligence. The major effort is to reduce the "how" instructions from human side. The combination of different technologies together is possible. A typical example is the application of AI methods to scheduling problem in [9]. However, the basic differences in mathematical tools are shown below.

For the DDC part, there are four fundamental mathematical tools: differential equation (DE), linear algebra (LA), complex variable analysis (CVA), and probability/statistics (PS). The DE and the LA construct the state equation approach. The DE and the CVA construct the frequence domain approach such as Bode Diagram analysis, root locus techniques, etc. The DE, the CVA, and the PS construct the Winner Filter Theory. The DE, the LA, and the PS construct the Kalman Filter Theory. With the sample-data concept and the digital controller design, automatic control with computers can be done easily.

For the FMS part, the key issue is the planning problem [10,11,12]. This was classified into strategic planning, tactical planning, and operational planning in [10]. At each level, some mathematical tools are used. For example, the CAN-Q model, mean value analysis, Petri nets, and perturbation analysis are used in the screening problem of strategic planning. Also this was classified into five categories in [11]: hierarchic framework, queueing theory models, integer programming models, simulation methods, and heuristic algorithms. A more general review involving economic justification and facility layout is in [12]. To sum up, the three essential mathematical tools are simulation, queueing theory, and mathematical programming.

For the CIM part, the effort is to achieve total automation. Many protocols were proposed to coordinate each subsystem such as MAP in factory communication network, GKS in computer graphics, etc. Theoretical works were also done. For example, the computational geometry in computer graphics involves wired-frame models, constructive solid models, boundary representation models, etc. The data base involves three types: relation, hierarchy, and network. However, total automation is not always the best solution for the maximum benefit of manufacturing system. Groover and Zimmer [13] gave the following formula for evaluation:

$$T_{lc} = T_1 + \frac{T_2}{Q} + \frac{T_3}{BQ}.$$

Here, T_{lc}, T_1, T_2, T_3, B, and Q are the average aggregate time, producing time, planning time, design time, the number of batches, and the number of units in each batch, respectively. For example, CAD/CAM is a way for partial automation. The mathematical tools involve network analysis, computational geometry, etc.

For the IMS part, many expert systems had appeared in the past years. Many AI methods were used such as heuristic search, knowledge representation, etc. Good efforts can be found in [14]. Recent research also involves uncertainty problem such as fuzzy set theory, Bayesian Theory, Dempster-Shafer Theory, etc. Using consistency analysis, all mathematical tools in this paper are listed in table 1. The column "Year" denotes the near time of the appearance of literature, not the time of the maturity of technology. The detail discussion of each mathematical tool is in the following sections.

CONSISTENCY ANALYSIS

From the intelligence in Turing way viewpoint, one of the most important features of IMS is the automatic reasoning ability of manufacturing system. Think about how a high automated manufacturing system works. Generally, a product request is specified by customer. The manufacturing procedure of this product request could be treated as a problem to be reasoned out by a manufacturing engineer. Then the engineer designs the procedure of producing the specified product. If the manufacturing system owns the reasoning ability, then we just show this specification to the underlying IMS. Next, the IMS will do reasoning and produce that product subsequently. From this viewpoint, we can replace the human factors by machines. A practical part machining example can be found in next section.

From the mathematical viewpoint, the underlying manufacturing system is a formal system. A most important property of formal system is the consistency analysis. If the underlying manufacturing system is a consistent system, then the reasoned out result can be used directly; otherwise, the user has to check whether there is any inconsistency in manufacturing system. Unfortunately, Godel gave his famous Incompleteness Theorem [8] in 1931 which limits the possibility of the consistency analysis for general systems. However, Godel's Incompleteness Theorem is a general theorem for general systems. If the domain of the underlying manufacturing system can be narrowed down, then this consistency analysis still can be done for some specific systems.

From this viewpoint, intelligent manufacturing systems based on the consistency analysis can be initially classified into Declarative Manufacturing Systems (DMS) [15], which are absolutely consistent manufacturing systems, and Descriptive Manufacturing Systems (D'MS) [16], which the consistency of manufacturing systems have to be checked or verified. Because the consistency of DMS is an inherent property of manufacturing system, the components of DMS can be declared arbitrarily and directly. In contrast, a designer of D'MS not only describe the characteristics of the underlying manufacturing system but also has to check the consistency. The overall classification of IMS based on consistency analysis is illustrated in Fig. 1. The classification of other parts in Fig. 1 is analyzed in subsequent sections.

RELATIONAL MANUFACTURING SYSTEMS

If the specified relations of the underlying manufacturing system are represented by atomic formulas [8] and relations are connected by connectives [8], then this type of manufacturing system is called a Relational

Table 1 A comparison of four categories of computerized manufacturing systems

Type	Year	Mathematical Tool	Example
DDC	mid-1950s	State Equation, Winner Filter, Frequency Domain Analysis, Kalman Filter	Numerical Control, PID controller
FMS	mid-1960s	Simulation, Queueing Theory, Mathematical Programming	AS/RS, AGVS, Parts Scheduling
CIM	mid-1970s	Computational Geometry, Algebra for Data Base, Network Analysis	CAD/CAPP/CAM, MAP/TOP
IMS	1980-	Mathematical Logic, Lambda Calculus, Touretzky's Theory, Category Theory	Expert Systems

G. R. Liang

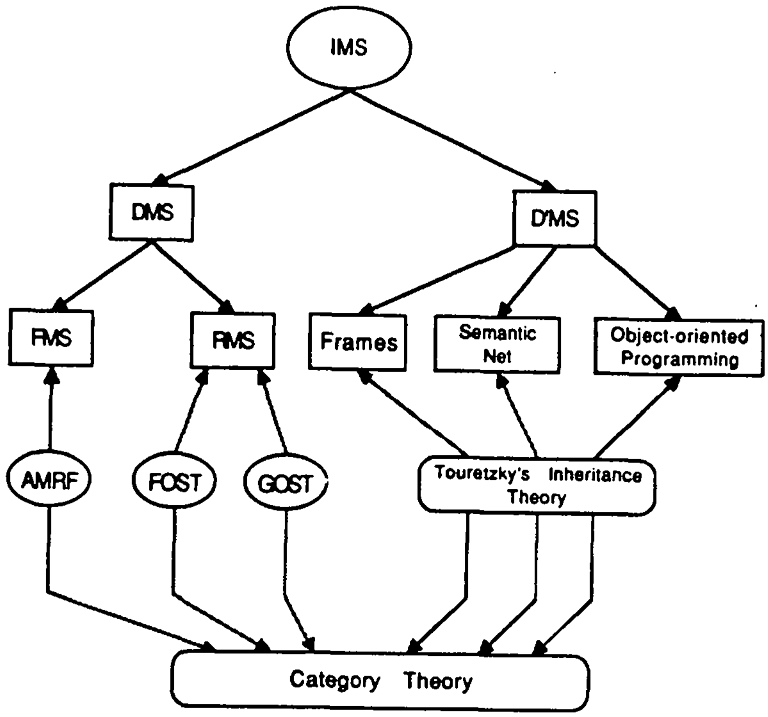

Fig. 1 The classification of IMS based on consistency analysis

Manufacturing System (RMS). An advantage of RMS is the reasoning by logic can be done easily. For example, think about how an NC programmer does his job. At first, he has to watch the surface of workpiece and reason out the way of tool motion, i. e. the tool paths in Fig. 2. Then he consults the handbook of NC programming, calculates the cutter locations, specifies the cutting parameters, and finds the proper instructions in order to control the NC machine tools.

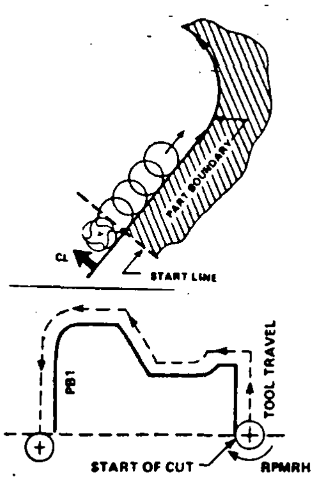

Fig. 2 Tool paths

Instead, an automated NC programming can be achieved using the intelligence in Turing way. First, the geometric features of tools can be specified as tool axioms. Second, the cutting parameters and computational formulas, i. e. the computation of cutter locations in Fig. 3, can be specified as the machining axioms and the computing axioms, respectively. Third, the ways of tool motion can be specified as *motion axioms*. Fourth, a raw workpiece and its finished workpiece are treated as the hypothesis and the conjecture to be proved, respectively. This reasoning-based automation model is shown in Fig. 4. The proving procedure is automatic, so is the cutting procedure. By the way, a generated proof is a part program for controlling the machine tool.

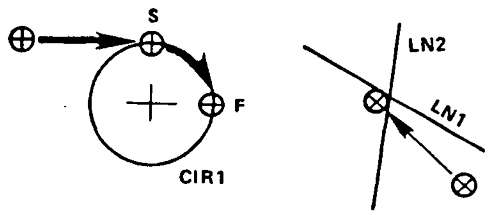

Fig. 3 Computing axioms

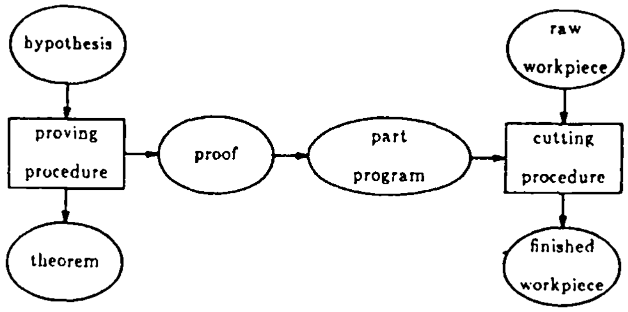

Fig. 4 The automation of a surface-generation problem

If all the well-formed formulas of axioms and hypothesis are limited to be definite clauses (D-clause) [17], then the underlying manufacturing system can be proved to be a consistent manufacturing system through the analysis of fixpoint semantics [17]. A successful implementation of RMS called the LINC system in automated NC programming problem can be found in [18]. The LINC system is classified into the Single Machine Problem (SMP). If there are many machine tools for surface generation, then this is called the Multiple Machine Problem (MMP). Several procedures were developed to compare their performance such as independent FOSTs approach, Combined FOSTs approach, GOST approach, etc [19]. Both SMP and MMP problems can be analyzed by the analysis of fixpoint semantics [19].

FUNCTIONAL MANUFACTURING SYSTEMS

A mathematical function will produce values, i. e. a data structure of a list of values, if all the arguments are given. Similarly, a factory will produce products if all the related inputs are given. If the functions of a manufacturing system can be treated as mathematical functions, then this type of manufacturing system is called a Functional Manufacturing System (F'MS). From a user viewpoint, a function of this factory is represented by a task command. So the physical functions of manufacturing system can be controlled through task commands. The execution of a task command will generate the desired output.

In operation, an F'MS is a hierarchical manufacturing system controlled by task commands. These task commands of F'MS can be assigned with input data into the underlying F'MS at any level to initiate the activities of F'MS. The execution of a task command at a level is equivalent to the decomposition of the main task into subtasks from current top level down to the lower levels. Then the subtasks in succession are downward repeatedly until all subtasks are done. For example, assign a task command to the extended AMRF factory [20] shown in Fig. 5. Assume the task command is assigned to the virtual cell level in Fig. 5. This task command can be decomposed into subtasks for workstations. Then subtasks are assigned again to robots, machines, buffers, etc. Similarly, any task command may be assigned arbitrarily to any level such as the shop level, the facility level, etc.

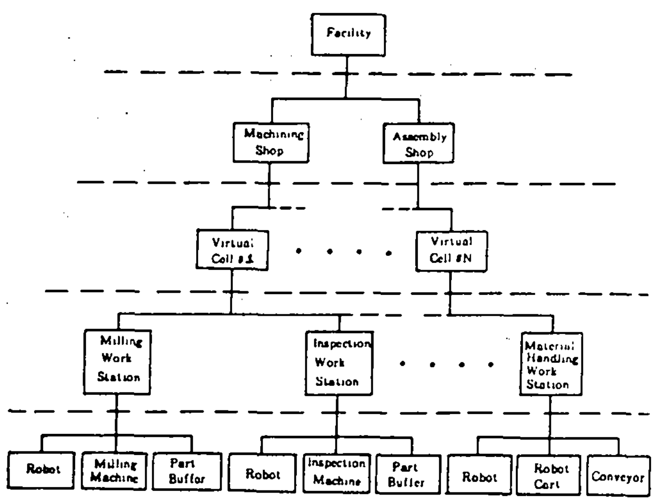

Fig. 5 The extended AMRF control hierarchy [20]

On analysis, a good mathematical model of F'MS is lambda calculus whose consistency was proved by D. S. Scott in 1969 [21,22]. Lambda calculus consists of λ-terms, axioms, and rules of inference. The λ-terms may be variables or terms generated by two basic operations on variables, say, abstraction and application. The set of λ-terms Λ is defined inductively in table 2. Some typical examples of λ-terms are shown below:

1. xx
2. λ x.xx
3. λ xy.yx(λz.z) (≡λx(λy((yx)(λz.z)))).

The axioms and the rules of the lambda calculus [21] are shown in table 3 and table 4, respectively. Through the axioms and the rules of lambda calculus, a λ-term can be reduced to normal form if this λ-term can not be reduced any more. For example, a λ-term (λx.λy.y)ab can be reduced to b by the β reductions below.

$$(\lambda x.\,\lambda y.y)ab \rightarrow_\beta (\lambda y.y)b \rightarrow_\beta b$$

Usually, there are many reduction methods [21] such as normal order reduction, applicative order reduction, head reduction, etc. Whatever reduction is chosen, the result is the same. Also, this procedure can be always done by normal order reduction. These good properties are implied by Church-Rosser theorems [21].

Table 2 The set of λ-terms

1. variable: x, y ε Λ
2. abstraction:
M ε Λ → $(\lambda x.M)$ ε Λ
3. application:
M, N ε Λ → (MN) ε Λ

Table 3 The axioms

axiom α: λx.E = λy.E[x←y],
y is not free in E.
axiom β: (λx.E)(F) = E[x←F].
axiom η: λx.Ex = E,
if x is not free in E.

Table 4 The rules

E=E
E=F → F=E
E=F, F=G → E=G
E=E' → FE = FE'
E=E' → EF = E'F
E=E' → λx.E = λx.E', rule ξ

In Fig. 5, a cell consists of workstations, and a workstation consists of physical components like robot, machine tool, etc. In fact, a cell is a collective noun. Its function depends upon its components and input data. For example, assume the cell #1 in Fig. 5 consists of three workstations: WS_1, WS_2, and WS_3. The corresponding λ-term representation is $Cell_1$ = X(WS_1)(WS_2)(WS_3), or $Cell_1$ (input-data) = f(WS_1,WS_2,WS_3,Input-data). Here, X is an operator on WS_i, i=1,2, or 3. Similarly, assume workstation WS_1 consists of robot R = $(\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x)$ and machine tool M = $(\lambda xy.x)(\lambda xy.x)$. Then assume WS_1 = YRM = $(\lambda xyz.xz(yz))((\lambda xy.x)((\lambda xyz.xz(yz))R))(\lambda xyz.xz(yz))MR)$.
If a task command is assigned to workstation WS_1 with input data $(\lambda x.x)(\lambda x.x)$, then the reduction is:

$$WS_1(\lambda x.x)(\lambda x.x) \rightarrow YRM(\lambda x.x)(\lambda x.x)$$
$$\rightarrow (\lambda xyz.xz(yz))((\lambda xy.x)((\lambda xyz.xz(yz))R))$$
$$(\lambda xyz.xz(yz))MR)(\lambda x.x)(\lambda x.x)$$
$$\rightarrow (\lambda xyz.xz(yz))((\lambda xy.x)((\lambda xyz.xz(yz))(\lambda z.z)))$$
$$(\lambda xyz.xz(yz))M(\lambda z.z))(\lambda x.x)(\lambda x.x)$$
$$\rightarrow \lambda x((\lambda xyz.xz(yz))(\lambda x.x)$$
$$((\lambda xy.x)x))(\lambda x.x)(\lambda x.x)$$
$$\rightarrow (\lambda xy.yx)(\lambda x.x)(\lambda x.x) \rightarrow \lambda x.x.$$

In the connection of the operations of F'MS with lambda calculus, each task command with meaning is corresponding to a λ-term which has head normal form (HNF) [21] because of the meaningless of λ-term of no HNF. Consequently, the capability of F'MS is represented by the set of useful task commands, so is the set of λ-terms which have normal forms. The decomposition of task is equivalent to the substitution of λ terms. The execution of a task command in F'MS is equivalent to the reduction sequence of a λ-term. So the activities of F'MS are represented

Table 5 The relationship between F'MS and lambda calculus

Lambda calculus in analysis	Operations in F'MS
λ-terms which have HNFs	task commands with meaning
set of λ-terms which have normal forms	system capability
the substitution process	the decomposition of task
the reduction processes	the activities of F'MS
the automation of reduction process	the automation of manufacturing
Church-Rosser Theorems	the uniqueness of command execution
λ-definability	computerized manufacturing

by the reduction processes. The automation of the reduction process is the automation of intelligent product manufacturing. The uniqueness of the execution result of a task command is implied by Church-Rosser theorems. The implementation of task command by computer is implied by λ-definability. In conclusion, the related relations are summarized in table 5. Some extensions had been done for analyzing the reconfiguration of manufacturing system in [23] and their topological properties in [24].

DECLARATIVE MANUFACTURING SYSTEMS

Both the F'MS and the D-clause-based RMS are consistent formal systems. Moreover, there are maybe the coexistence of functional and relational relations in a DMS. From this viewpoint, there are two kinds of DMS [15]: concrete model and abstract model. For the abstract model approach, the functions and relations can be combined together. A good reference for discussing their properties is [25].

For the concrete model approach, a DMS consists of subsystems, i. e. RMS and F'MS. According to the ways of combination, three types of DMS are classified out: type 1, type 2, and type 3 [15]. One example of type 1 DMS is the system of robot arm and machine tool. The protocol system was implemented at NCTU [15].

DESCRIPTIVE MANUFACTURING SYSTEMS

A D'MS is a manufacturing system owning the reasoning ability, but it may be an inconsistent system at the same time. For example, some manufacturing systems are described by frames [26,27], object-oriented programming [28], or semantic nets [29]. A common problem to produce the inconsistency is the property of inheritance. For example, different reasoning consequences result from changing the key-in sequence of the description of manufacturing system. A case is the use of the override instructions in expert system KEE [16].

A good way to solve the consistency problem is through Touretzky's inheritance theory. From Touretzky's viewpoint [29,30], the set of inheritance assertions is similar to the axioms of a formal system. They are basic assumptions, and all the conclusions could be generated from them. The rule of inference is the inferential distance ordering. The inheritance path is equivalent to the proof of a theorem. The correspondence of theory in logic is the grounded expansion. The consistent part of manufacturing system can be found if the set of inheritance assertions is consistent. A brief summary of Touretzky's inheritance theory is listed in table 6.

Another example of D'MS is the equational calculus approach in the design of Production Management Language (PML). A good reference is in [31].

Table 6 A comparison of Touretzky's formal system with the theory of mathematical logic

Touretzky's Formal System	Mathematical Logic
inheritance assertions	axioms
inferential distance ordering	rule of inference
inheritance path	proof
conclusion set	set of theorems
grounded expansion	theory
ambiguity	model

CATEGORY-THEORETIC FOUNDATION

In the previous sections, predicate calculus, lambda calculus, and Touretzky's inheritance theory/equational calculus are applied to modeling RMS, F'MS, and D'MS, respectively. Surely a unified theory for integrating all approaches is expected. Here, a category-theoretic foundation of IMS, shown in Fig. 6, is conjectured for four reasons in the following.

First, the fixpoint semantics of RMS can be analyzed by Kleene Fixed Point Theorem in category theory [32]. A more detailed correspondence between the lattice-theoretic approach and the category-theoretic approach can be found in [32]. Moreover, the MMP can be solved in a similar way.

Second, the mathematical model of F'MS is lambda calculus. Lambek [33] gave a closed relationship between lambda calculus and Cartesian-Closed Category (CCC). So the analysis of F'MS can be done by CCC.

Third, the properties of Touretzky's inheritance theory is similar to the properties of formal system in mathematical logic [8]. Also a closed relationship between mathematical logic and category theory can be found in [34]. However, a rigorous proof is still absent.

Fourth, the initial semantics of equational calculus can be manipulated by category theory [35].

If the relationship in the third reason can be proved to be true, then the analysis of IMS can be done easily by category theory. For ex-

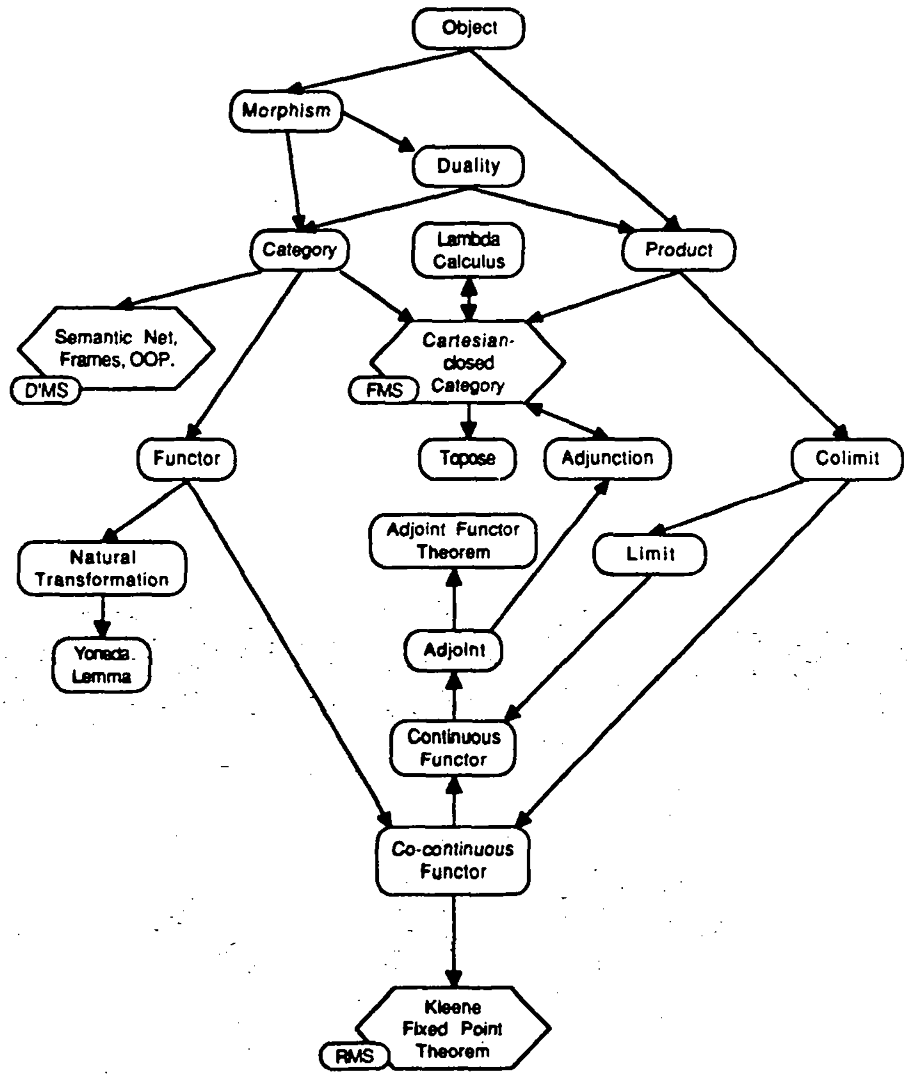

Fig. 6 The category-theoretic foundation of IMS

ample, because different mathematical tools are used, there is no way to analyze an IMS which consists of F'MS, RMS, and consistent D'MS. But this problem would be an easy job if there exists a category-theoretic foundation of IMS.

CONCLUSION

Through the consistency analysis, IMS can be classified into F'MS, RMS, DMS, and so on. Each system can be analyzed in detail using some mathematical tools such as predicate calculus, lambda calculus, etc. Clearly, a unified theory for integrating different approaches is expected. An effort is the category theory as the foundation of IMS. However, the proposition is completed if a rigorous proof in the Touretzky's part can be found.

ACKNOWLEDGEMENT

The author would like to thank professor J. M. Proth and Mr. J. Hilger, project SAGEP, INRIA-Lorraine, France, for the suggestions on the equational calculus in this paper.

REFERENCES

[1]. Mille, R. K. and Walker, T. C., Artificial Intelligence in Manufacturing, SEAI Technical Publication, 1988.

[2]. Kumara, S. T., Kashyap, R. L., and Soyster, A. L., Artificial Intelligence, Manufacturing Theory and Practice, Industrial Engineering and Management Press, IIE, 1989.

[3]. Liang, G. R., "AI and Intelligent Manufacturing Systems," Journal of Mechatronic Industry, pp. 148-160, Dec. 1987.

[4]. Oliff, M.D., Expert Systems and Intelligent Manufacturing, North-Holland, 1988.

[5]. Minsky, M. L. and Papert, S. A., Perceptrons, Expanded Edition, The MIT Press, 1988.

[6]. Turing, A. M., "Computing Machinery and Intelligence," (Feigenbaum, E. and Feldman, J., Ed.) Computers and Thought, pp. 1-35, McGraw-Hill, New York, 1963.

[7]. Charniak, E. and McDermott, D., Introduction to Artificial Intelligence, Addison-Wesley Publishing Company, 1985.

[8]. Mendelson, E., Introduction to Mathematical Logic, D. Van Nostrand Company, 1979.

[9]. Fox, M. S. and Smith, S. F., "ISIS - a Knowledge-based System for Factory Scheduling," Expert Systems, Vol. 1, No. 1, pp. 25-49, 1984.

[10]. van Looveren, A. J., Gelders, L. F., and van Wassenhove, L. N., "A Review of FMS Planning Models," (Kusiak, A., Ed.) Modeling and Design of Flexible Manufacturing Systems, pp. 3-31, North-Holland, Amsterdam, 1986.

[11]. O'Grady, P. J. and Menon, U., "A Concise Review of Flexible Manufacturing Systems and FMS Literature," Computers in Industry, pp. 155-167, 1986.

[12]. Kalkunte, M. V., Sarin, S. C., and Wilhelm, W. E., "Flexible Manufacturing Systems: a Review of Modeling Approaches for Design, Justification, and Operation," (Kusiak, A., Ed.) *Flexible Manufacturing Systems:* Methods and Studies, pp. 3-25, North-Holland, 1986.

[13]. Groover, M. P. and Zimmers, E. W., CAD/CAM Computer-aided Design and Manufacturing, Prentice-Hall, 1984.

[14]. ERC Posters Presented at the Annual Research Forum, Engineering Research Center for IMS, Purdue University, Sept. 23, 1986.

[15]. Liang, G. R., "Declarative Manufacturing Systems," Journal of Mechatronic Industry, pp. 246-254, July, 1988.

[16]. Liang, G. R., "Expert System KEE and Descriptive Manufacturing Systems," Journal of Mechatronic Industry, pp. 100-112, Aug. 1989.

[17]. van Emden, M. H. and Kowalski, R. A., "The Semantics of Predicate Logic as a Programming Language," JACM, 23, 4, pp. 733-742, Oct. 1976.

[18]. Liang, G. R. and Liu, C. R., "Logic Approach to Surface Generation (1): Single-Machine Problem," ASME Winter Annual Conf., Intelligent and Integrated Manufacturing Analysis and Synthesis, Boston, Dec. 1987.

[19]. Lu, S. F., Logic Approach to Process Plans Generation, Master Thesis, Institute of I. E., National Chiao Tung University, 1989.

[20]. Jones, A. T. and McLean, C. R., "A Proposed Hierarchical Control Model for Automated Manufacturing Systems," Journal of Manufacturing Systems, Vol. 5, No. 1, pp. 15-25, 1986.

[21]. Barendregt, H. P., The Lambda Calculus - Its Syntax and Semantics, North-Holland, 1984.

[22]. Liang, G. R., "Functional Manufacturing Systems," Journal of Mechatronic Industry, pp. 182-189, May, 1988.

[23]. Liang, G. R., "The Reconfiguration of Manufacturing Systems Using Transformation Methods," Proceeding of the 3rd National Workshop on Automation Technology, Kaohiung, Taiwan, July 1989.

[24] Cheng, Y. Y. and Liang, G. R., "The Analysis of Functional Manufacturing Systems Using Bohm Trees," Proceeding of the 3rd National Workshop on Automation Technology, Kaohiung, Taiwan, July 1989.

[25]. Bellia, M. and Levi, G., "The Relation Between Logic and Functional Languages: a Survey," J. Logic Programming, Vol. 4, pp. 217-236, 1986.

[26]. Huber, A., "Knowledge-based Production Control for a Flexible Flowline in a Car Radio Manufacturing Plant," Proc. 6th Int. Conf. Flexible Manufacturing Systems, pp. 3-19, Nov. 1987.

[27]. Weber, D. M. and Moodie, C. L., "A Knowledge-based System for Information Management in a Automated and Integrated Manufacturing Systems," Robotics & Computer-integrated Manufacturing, Vol. 4, No. 3/4, pp. 601-617,1988.

[28]. Lin, J. T. and Yen, K. J., "Objected-oriented Simulation of FMS," Proceeding of the 2nd National Workshop on Automation Technology, pp. 131-158, Kenting, Taiwan, April 1988.

[29]. Touretzky, D. S., The Mathematics of Inheritance Systems, Pitman Publishing Limited, 1986.

[30]. Wang, S. M. and Liang, G. R., "The Analysis of Descriptive Manufacturing Systems Using Touretzky's Inheritance Theory," CIIE Annual Conf., Hsinchu, Taiwan, Nov. 1989.

[31]. Hilger, J., Production Management Language and Proof System, INRIA Research Report, France, 1990.

[32]. Liang, G. R., "Category-theoretic Approach to the Semantics of FOST," Proceeding of the 2nd National Workshop on Automation Technology, pp. 527-534, Kenting, Taiwan, April 1988.

[33]. Lambek, J., "From λ-calculus to Cartesian Closed Categories," (Hindley, J. R. and Seldin, J. P., Eds.), To H. B. Curry: Essays on Combinatory Logic, Lambda-Calculus and Formalism, Academic Press, pp. 375-402, 1980.

[34]. Goldblatt, R., Topoi, the Categorical Analysis of Logic, North-Holland, 1984.

[35]. Ehrig, H. and Mahr, B., Fundamentals of Algebraic Specification - Equations and Initial Semantics, Springer-Verger, 1985.

ISSN 0249-6399